LA-UR-18-22727
Approved for public release; distribution is unlimited.

Title: ExtendSim Scenario Manager Overview

Author(s): Johnson, Joel Riding
Psaila-Dombrowski, Maureen Justine

Intended for: Report

Issued: 2018-03-29

LA-UR-18-XXXXXX
Approved for public release; distribution is unlimited.

ExtendSim Scenario Manager Overview

Joel Johnson, AET-2

Maureen J. Psaila-Dombrowski, AET-2

Revision 0: March 27, 2018

Table of Contents

Introduction

ExtendSim® is an advanced simulation environment used to model processes using either discrete event, continuous, discrete rate or agent-based modeling techniques. The process models created are comprised of blocks that are linked together to simulate the logic and sequence of the process being evaluated. These blocks can communicate with each other via messages. This ensures that required information is communicated throughout the model as needed. In addition, ExtendSim has a number of data management techniques, including integrated databases that can be used to store input and output information from the model.

When creating a model, there is often variability in the input data. For instance, any activity in the model may have a range of completion times or perhaps the number of items involved in the activity may vary. This means that there is rarely a single answer from the process model. In order to assess process performance accurately, the impact of the variation in input data should be evaluated. As the process to be modeled becomes more complex, the number of scenarios that must be created to evaluate the process variability increases dramatically.

ExtendSim has a specific block designed to help with this complex scenario analysis, the Scenario Manager. The Scenario Manager provides an interface to explore the modeling parameters used and is able to capture the results of the modeling process in ExtendSim's integrated databases. This allows for a systematic approach to evaluating a process. **Factors** are the model inputs, which are assigned and varied and used to create multiple **scenarios**. These scenarios are used to record and evaluate how the model reacts to changes in inputs through **responses**. Each scenario is created using different factors that vary between scenarios, and all scenarios are run multiple times to capture stochastic behavior. The scenarios can be varied using a formal **design of experiment** (DOE) analysis methodology or in a more informal manner. The Scenario Manager allows the users to establish the scenarios ahead of time and perform the evaluation without further input. Table 1 below defines the terminology used in the Scenario Manager.

Scenario analysis is a powerful tool for developing a thorough understanding of process dynamics and comparison of alternatives. Among other things, it can be used for

- Systemically and strategically examining the outcome of different model configurations,
- Understanding process behavior and performances,
- Evaluating process robustness, and
- Evaluating process alternatives.

This document is intended as a guide to be used as a supplement to the section in the ExtendSim User Guide covering the Scenario Manager.[1] This guide will provide further explanations and examples to elucidate the use of scenario analysis and the Scenario Manger block.

[1] *ExtendSim v9 User Guide*. San Jose, CA (2013), Imagine That Inc,. pp 634 - 649.

Table 1: Scenario analysis terminology. The Scenario Manager uses a standard design of experiments (DOE) nomenclature.

Term	**Definition**
Factor	An input parameter that affects the model. May be specified using dialogue parameters or a database. Factors may have a range of values. *Example: Varying how many cashiers are available in a grocery store in a Resource Pool.*
Response	The simulation result obtained by the model for each scenario. The user can choose which responses are recorded for each scenario. Responses can be recorded in the Scenario Manager block or a database. *Example: Average number of customers who waited in each Queue or average wait time per customer in a Queue.*
Scenario	A scenario is a set of factors that are input in a model during setup. A scenario may have any number of factors. *Example: Comparing the effects on Queue length and average wait time between using three cashiers in a Resource Pool compared with four cashiers.*
Design of Experiment	A systematic method used to determine the relationship between factors affecting a model and the responses of that model.
Number of Runs	Number of times the model is run for each scenario (i.e. using the same factors).

Scenario Manager Block

As mentioned previously, the Scenario Manager Block is a data storage and analysis system that allows the user to systematically vary the inputs (factors) of a model to create different scenarios, run each scenario the desired number of times, and record the output of the model for each scenario (responses). This section provides a brief overview of the block.

The Scenario Manager block is found in the "Value" library (see Figure 1). The block has no connectors and can be placed anywhere in the model.

Figure 1: Scenario Manager Block.

Figure 2 show the tabs found in the Scenario Manager block:

- **Factors (Model Inputs):** The factors tab allows the user to identify the factors used in the model. The location of each factor within the model, range of values allowed for factors, and the step size used in the evaluation can be specified here. This information can be entered in the dialog box provided or by using database factors.
- **Responses (Model Results):** The responses tab identifies the responses, or results, from the model. The option to include or not include responses in the report is available here. Responses can be specified in the dialog responses table that are written to the scenarios tab or pulled from and saved to a database using a database response.
- **Scenarios:** The scenario tab facilitates the creation of specific scenarios to be evaluated. These scenarios are specified by the values of the factors used for the scenario. Responses for each scenario can be recorded in this tab. In addition, the run details are indicated in this tab including the number of runs for each scenario and the simulation start and end times. Factors and factor values are specified for each scenario before the model is run and responses to be recorded are specified (all scenarios record all responses); while the response values are recorded after each scenario has run.
- **Export:** The export tab allows the user to export a response report in a variety of formats. The report type, file type and location are selected here. The user can export the complete results, the displayed statistics or the complete statistics.
- **Comments:** The comments tab allows the user to document comments that are only saved in the Scenario Manager.

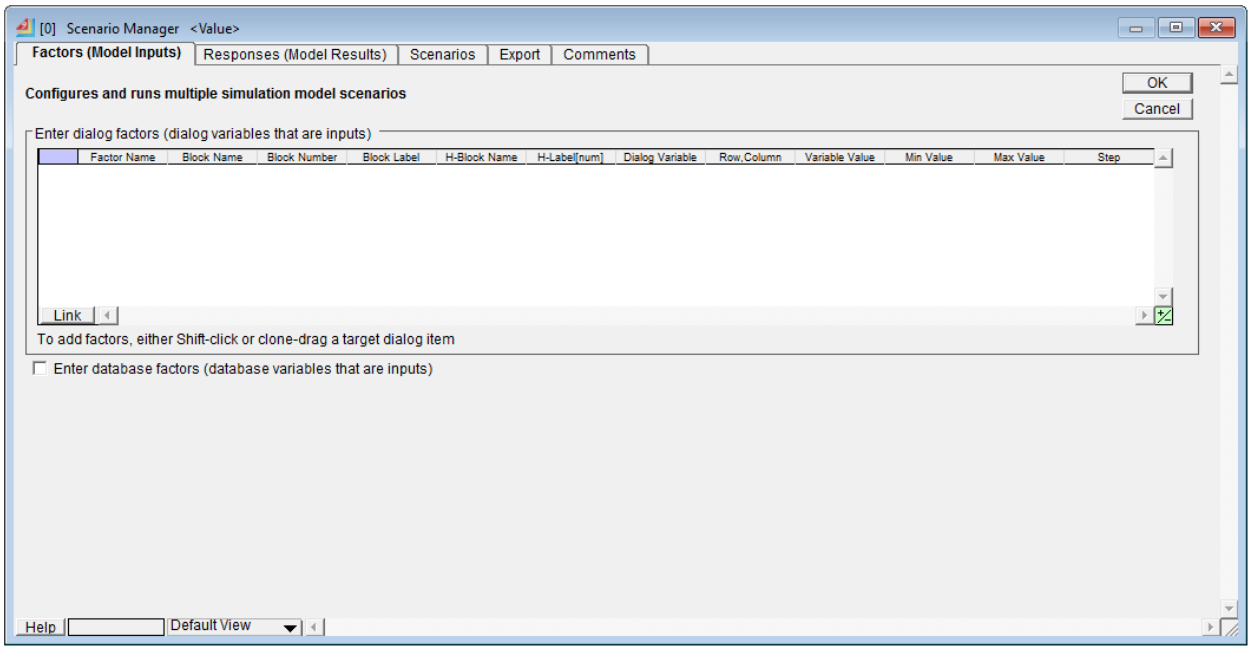

Figure 2: The Scenario Manager window.

How to Use the Scenario Manager

There are a number of steps that must be performed in order to execute a scenario analysis. After a process model is built and ready to be run, the user should follow these general steps:

- Setup
    - Add a Scenario Manager to the model
        - This block may be added anywhere in the model
    - Identify and add the factors (model inputs) needed
    - Identify and add the responses (model results) needed
    - Create scenarios
        - Determine DOE and add scenarios to the table
        - Determine necessary number of runs for each scenario
- Run the model
- Extract the data
- Analyze the results

This guide will focus the setup, running the model, and data extraction steps. Analyzing results should be completed using best practices involved with statistical analysis.

Setup

Once the Scenario Manager is added to the model, factors, responses and scenarios must be included in the block in each respective tab.

Factors (Model Inputs) Tab

Model inputs are added and edited here. This step should be completed after the model is created and input ranges are known. At least one factor is required to run the Scenario Manager. Factors can be a variable found in a block or a database, table, field, or record. These are divided into two types of model inputs: dialog factors and database factors. Instructions for adding both are found below. Figure 3 shows both methods implemented in a single Scenario Manager block.

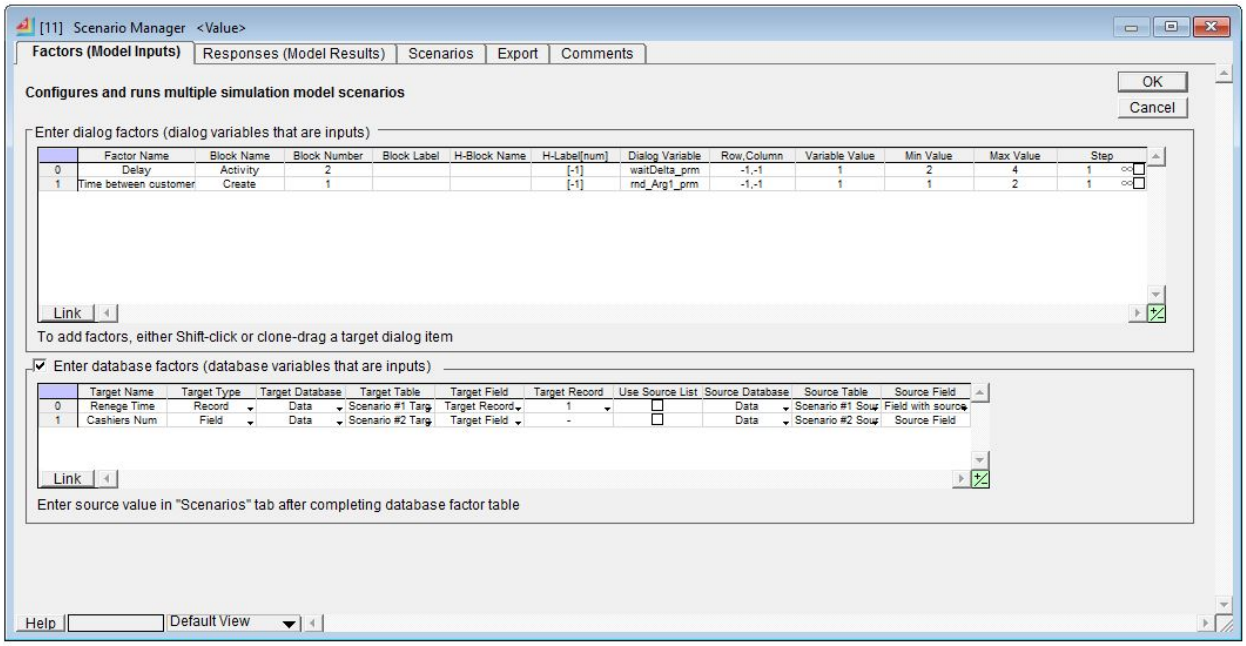

Figure 3: The factors tab with dialog factors in the first table and database factors in the second table.

Dialog Factors

Dialog factors are the first type of factor used when building scenarios. It is named as such because they are added through dialog boxes within model blocks. Dialog factors can be viewed, edited, and deleted in the Factors tab in the "Enter dialog factors" table. A description of the columns of the dialog factors spreadsheet is provided in Table 2. The dialog factors are added in block dialog windows using one of three methods:

Shift-Click method

1. Open the block that contains the factor you want to add to the Scenario Manager. Figure 4 shows the process dialog for an activity block.
2. Hold the Shift key and select the input box containing the desired factor and select "Scenario Manager: Add Factor"
3. A pop-up window will appear to name the factor. Give the factor a name to be used in the Scenario Manager.
4. Press the OK and this value will appear in dialog factors table in the Scenario Manager.

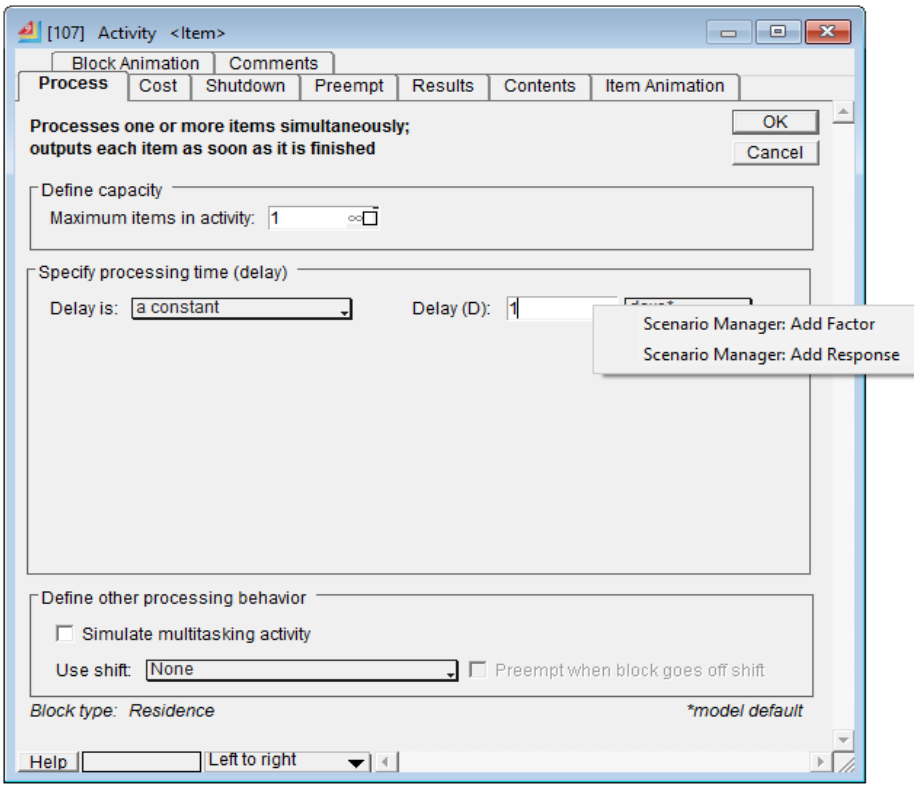

Figure 4: An Activity block window showing the option to add the Delay as a factor. This menu is accessed by Shift+Left Click in the dialog box.

Clone-Drop method

Another method that can be used to enter dialog factors is the Clone-Drop method.

1. Select the Clone arrow (the arrow in top bar with a bullseye attached – a circle surrounding a dot) or right-click on the parameter you want to select and choose the Clone Tool option.
2. In a block's dialog window, drag the desired input box and drop it into the Scenario Manager's icon.
3. In the popup window that appears, select Factor.
4. This value will appear in dialog factors table in the Scenario Manager.

Manual Method

Dialog factors can be manually entered into the Scenario Manager by typing the variable name into the factors table of the block.

A step-by-step tutorial for using dialog factors using the "Shift-Click" method can be followed in the User's Guide using Tutorial 1 "Final Car Wash Scenarios."[2] Table 2 reviews each of the 12 columns in the "Enter dialog factors" table. Rows in this table should be updated if a block containing a factor is edited or deleted.

[2] *ExtendSim User Guide*. San Jose, CA (2013), Imagine That Inc., pp 635 - 639.

Table 2: Description of dialog factors table (see Figure 3).	
Table Header	**Description**
Factor Name	The “Factor Name” is an identifier chosen by the user. It is used as a column header in the Scenarios table and in any exported reports.
Block Name	The name of the block containing the dialog variable, most often the type of block.
Block Number	The number of the block containing the dialog variable. The block name, label, hierarchical block name, and hierarchical block label are determined from the block number. The block number appears in the title bar of a block's dialog. In the case where the block is used inside of a hierarchical block, there are two numbers in the title bar of the block’s dialog, the first number is the global block number (which is used here), the second value is the block’s number inside of the hierarchical block.
Block Label	The label of the block that contains the dialog variable.
H-Block Name	The name of the hierarchical block, if the dialog variable is in a block inside of a hierarchical block.
H-Label[num]	The local number of the block within the hierarchical block, if the dialog variable is in a block inside of a hierarchical block.
Dialog Variable	The name of the dialog variable used in the scenario analysis.
Row, Column	The row and column values of that parameter in the table if it is in a table
Variable Value	The current value of the parameter.
Min Value	The minimum value that the parameter can take in the model – this is user specified.
Max Value	The maximum value that the parameter can take in the model – this is user specified.
Step	The number of points that the parameter can take between the minimum and maximum values of that parameter – this is user specified. The range for each factor is used for the automatic generation of DOE scenarios.

Database Factors

Database factors are the second type of factor, and are named because the model must read input values from an ExtendSim database to be able to implement database factors. Database factors have an advantage over dialog factors because they can be easily varied (such as entire databases, tables, fields and records used as a single factor).

Database factors can be accessed on the lower half of the Factors tab as shown in Figure 5 and are enabled when the "Enter database factors" box is checked. Database factors may be added in one of two ways: manually or through the "Link" button. Both methods have a source database element and a target database element. The source database reference contains the values that will be swapped out in between each scenario, and the target database is the database, table, field or record read by the model. Typically, the target database is used by the model to read values for block parameters, item/value creation, or attributes. A database must be created and set up before being used in the scenario manager. At the start of each set of scenarios, the Scenario Manager copies the contents of the source database factor variable into the target database factor variable.

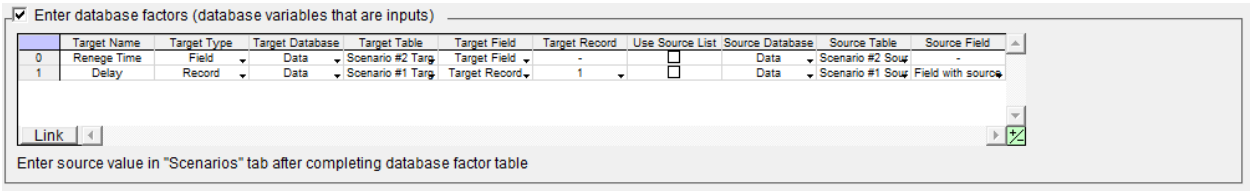

Figure 5: The enabled database factors table.

Manual Method

To add database factors, manually resize the table to include the number of desired database factors by clicking the small +/- sign in the lower right hand corner of the table. Once rows are added to the table, each column's information is manually filled out. Table 3 describes each column's function and how the information is inputted into that column.

Table 3: Description of database factors table (see Figure 5).

Table Header	Description
Target Name	A text cell that assigns a name identifier to the factor used in the Scenarios tab. Information can be entered manually or is automatically created.
Target Type	A drop down menu that allows you to select the target type (database, table, field, or record).
Target Database	A drop down menu that allows you to select the name of the target database. This is the database that the model currently uses to read block parameters. At the start of each scenario, the source data for that scenario is entirely copied into the target database.
Target Table	A drop down menu that allows you to select the name of the target table. If the target type is database, the target table column is not used.
Target Field	A drop down menu that allows you to select the name of the target field. If the target type is database or table, the target field column is not used.

Target Record	A drop down menu that allows you to select the name of the target record. If the target type is database, table, or field the target record column is not used.
Use Source List	If "Use Source List" is selected, the source database, table, and field (as listed below) will contain a list of possible source values. This option is required if you plan to use the design of experiments feature. This option is recommended for target types of databases and tables. The "Use Source List" option is not used if the target type is field or record.
Source Database	A drop down menu that allows you to select the database that contains the source values for the scenarios.
Source Table	A drop down menu that allows you to select the table that contains the source values for the scenarios.
Source Field	A drop down menu that allows you to select the field that contains the source values for the scenarios. If the "Use Source List" option is checked, this will contain a list of the possible sources for the target.

Link Method

Linking the database responses table to a database table allows the user to add multiple responses simultaneously. To set up a database table for this purpose, ten fields must be present in the database table. Neither the table nor the fields have name requirements, but certain fields have field type requirements. The "Target Name" and "Use Source List" may be either strings or numbers, while the rest of the fields must be set as strings. For the "Use Source List" column, records must contain any number besides zero (positive or negative) to check the box true or the number zero or any string to leave the box unchecked (false). All other field types must be set to String and must contain a valid label. For instance, the second field which corresponds to "Target Type" must contain one of the possible target types: data, table, field, or record. "Target Database," "Target Table," "Target Field," "Source Database," "Source Table," and "Source Field" must all contain the name of an actual database, table, field, and record. If a cell should be empty, then the cell may be left blank or contain a hyphen.

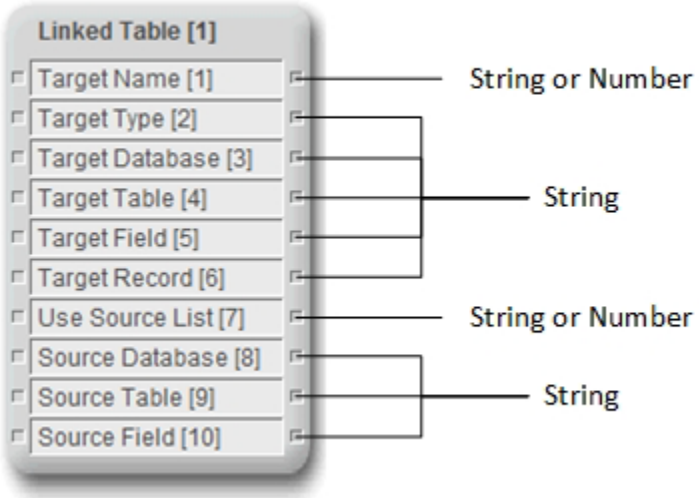

Figure 6: Field type requirements for linking a database to input database factors.

To delete a linked database, either delete the row with the database factor or delete the table, which was linked in the database viewer.

Note: If database factors were added manually and then the table was then linked to a database table, an array error will arise if the number of rows of manually added database factors is more than the number of rows in the table linked even though the manually added factors are no longer in use.

To correctly setup database factors, the source database and target database must be properly formatted. A description of all required information is provided in Table 3. See example of adding database factors to "Final Car Wash Scenarios" in the ExtendSim User Guide.[3] To illustrate how the database factors might look in the factors table, three examples are presented here. The database used in these examples is shown in Figure 7 , with each example labeled with the type of factor being employed.

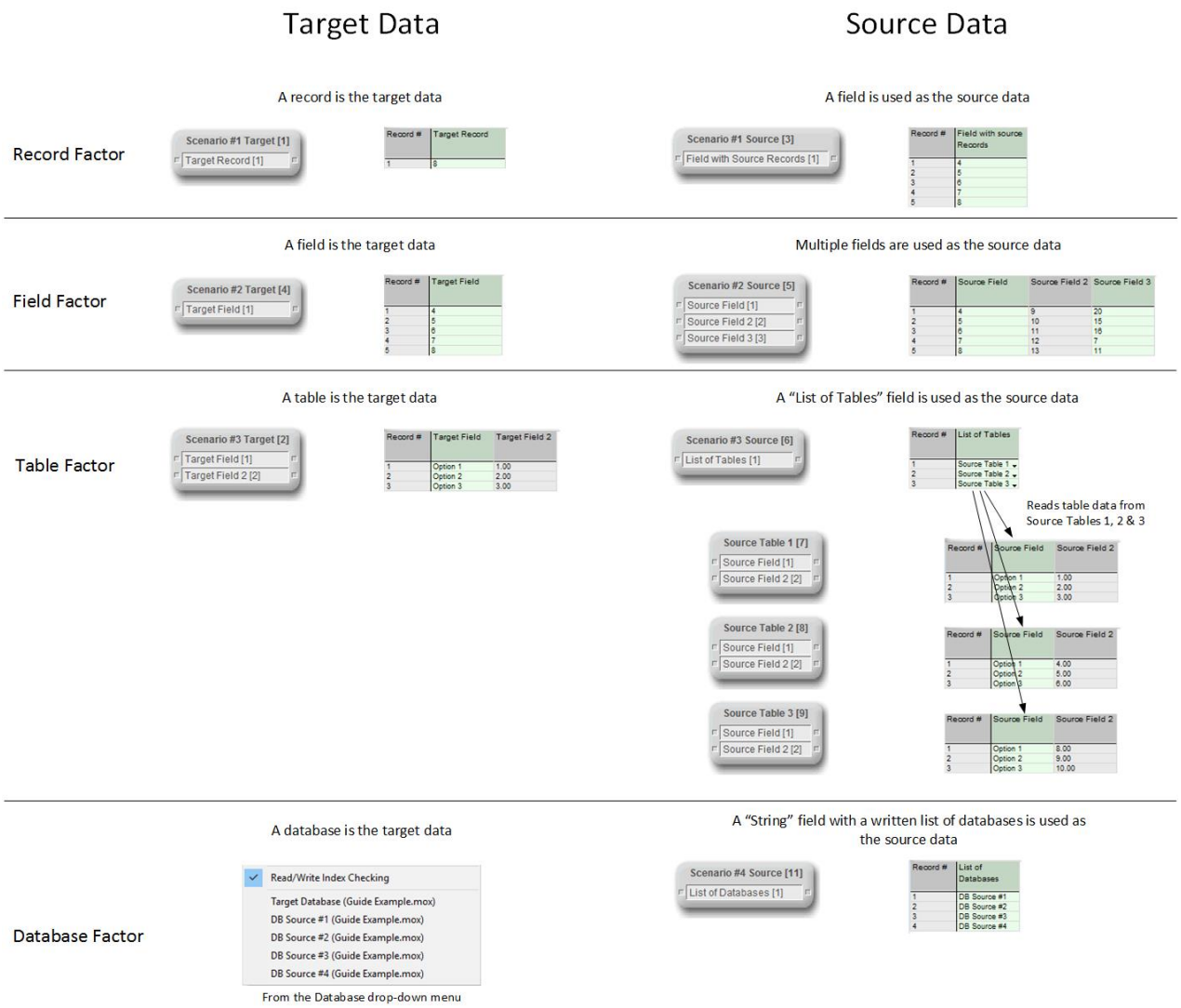

Figure 7: The database setup to add a record, field, and table as a database factor.

[3] *ExtendSim User Guide*. San Jose, CA (2013), Imagine That Inc., pp 642 - 647.

The first example illustrates using a **record** database factor. Examples of how a database record may be used in a model are linking a Create block or Item/Value Schedule block to a record in a database or by calling the value in an equation block. A field (column) with multiple rows is used as the source. The Scenario Manager switches the target record with each row's value until each row has been used. Table 4 shows how the database record would be formatted in the table using the example from Figure 7.

Table 4: Example table showing how to add a record as a database factor.

Target Name	Target Type	Target Database	Target Table	Target Field	Target Record	Use Source List	Source Database	Source Table	Source Field
Target Record	*Record*	*Default Database*	*Scenario #1 Target*	*Target Record*	*1*	☐	*Default Database*	*Scenario #1 Source*	*Field with source Records*

The second example uses a **field** factor. This is similar to using a record as a factor, the only difference being the source table must have multiple fields, which are used to replace the target field in each scenario. Table 5 shows how the database record would be formatted in the table using the example from Figure 7.

Table 5: Example table showing how to add a field as a database factor.

Target Name	Target Type	Target Database	Target Table	Target Field	Target Record	Use Source List	Source Database	Source Table	Source Field
Target Field	*Field*	*Default Database*	*Scenario #2 Target*	*Target Field*	-	☐	*Default Database*	*Scenario #2 Source*	-

A **table** factor may also be used as a factor in a model. Step-by-step instructions are found on pages 642 to 647 of the ExtendSim v9 User Guide. Table 6 shows an example setup for a table factor. Please note that the source field refers to a field with multiple records in rows, each record is a reference to different tables within the database. In the User Guide PDF, the figure on page 646 shows the structure of the model data needed to use a table as a factor. Table 6 shows how the database record would be formatted in the table using the example from Figure 7.

Table 6: Example table showing how to add a table as a database factor.

Target Name	Target Type	Target Database	Target Table	Target Field	Target Record	Use Source List	Source Database	Source Table	Source Field
Target Table	*Table*	*Default Database*	*Scenario #3 Target*	-	-	☒	*Default Database*	*Scenario #3 Source*	*List of Tables*

A **database** factor is the last type of factor. Table 7 shows an example setup of using a database factor. Please note that the source field refers to a field with multiple records with each record being a name of one of the databases in the model. Unlike the table factors, these are input by hand, as the field is set as a string.

Table 7: Example table showing how to add a database as a database factor.

Target Name	Target Type	Target Database	Target Table	Target Field	Target Record	Use Source List	Source Database	Source Table	Source Field
Target Table	*Database*	*Default Database*	-	-	-	☒	*Default Database*	*Scenario #4 Source*	*List of Databases*

Figure 8 shows how to create the source database table formatted as a list of tables. Each record in this field is individually selected as the tables that will be swapped in the target database.

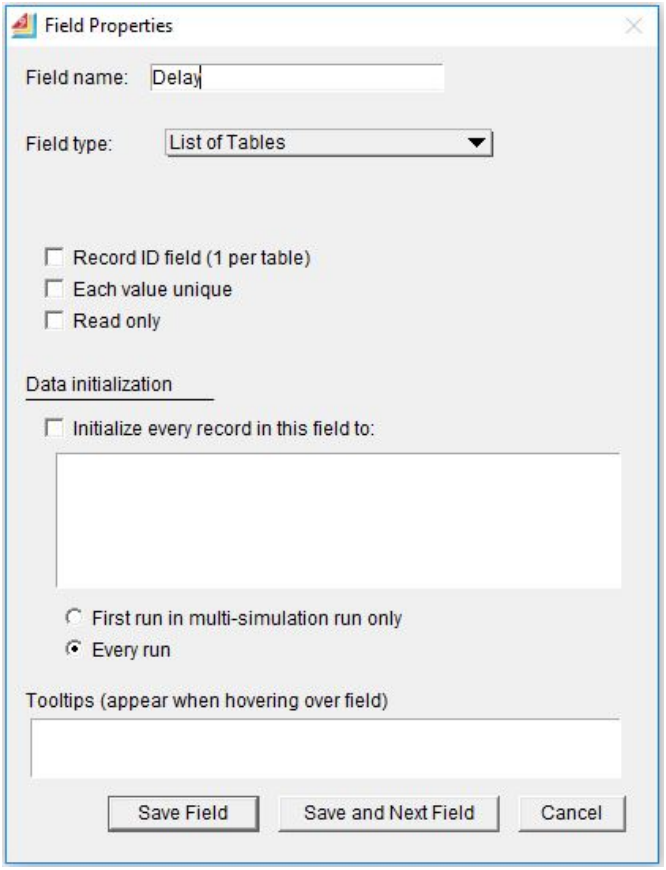

Figure 8: When adding a table as a database factor, create a field as the type "List of Tables".

Responses (Model Outputs)

Model outputs are identified and edited in the Responses tab of the Scenario Manager block. This step should be completed after the model is capable of running and has results that can be captured. Any number of responses may be added to the Scenario Manager. There are two types of responses that may be added. **Dialogue responses** are listed in the first table found on this tab. **Database responses** are listed in the table that appears after checking the "Enter database responses" box. Figure 9 shows a Scenario Manager view of both types of responses used in a model.

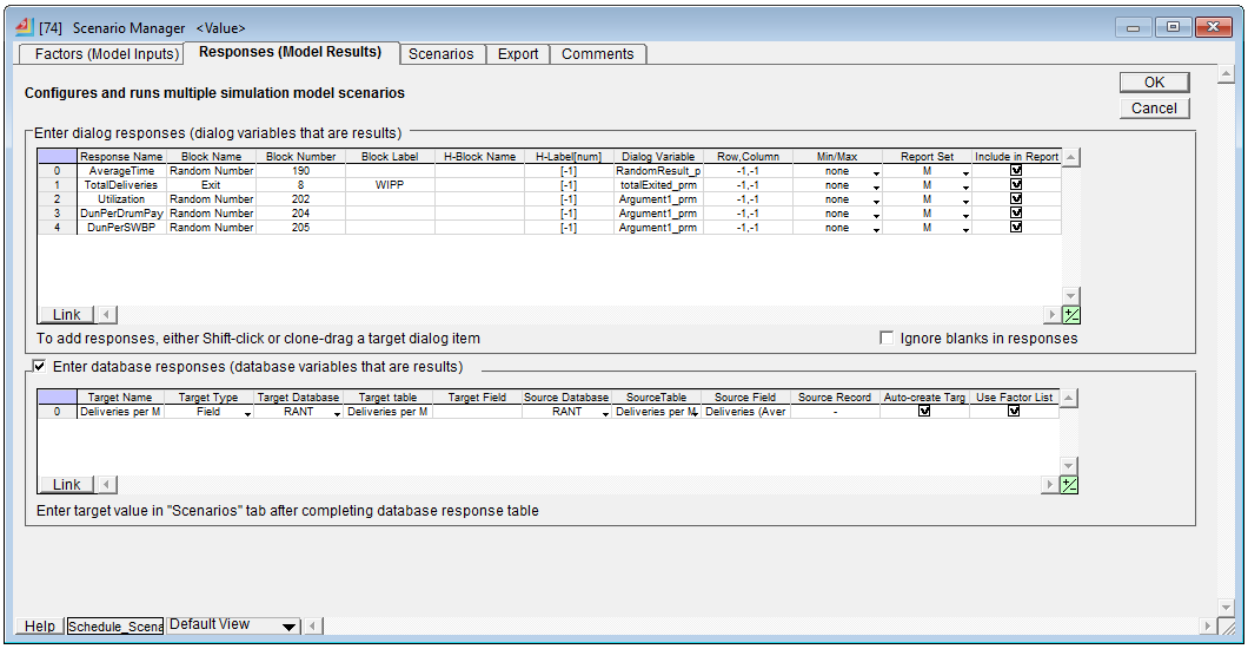

Figure 9: The responses tab with dialog responses in the first table and database responses in the second table.

Dialog Responses

Dialog responses can be viewed, edited, and deleted in this tab in the "Enter dialog responses" table. These factors are added in block dialog windows using one of the following methods:

Shift-Click Method (see Figure 10)

1. Open the block dialog interface and place the cursor over the response needed.
2. Hold the Shift key and select the dialog box containing the desired response (Shift + click).
3. Select "Scenario Manager: Add Response"
4. A pop-up window will appear to name the response.
5. Press the OK and this value will appear in dialog factors table in the Scenario Manager.

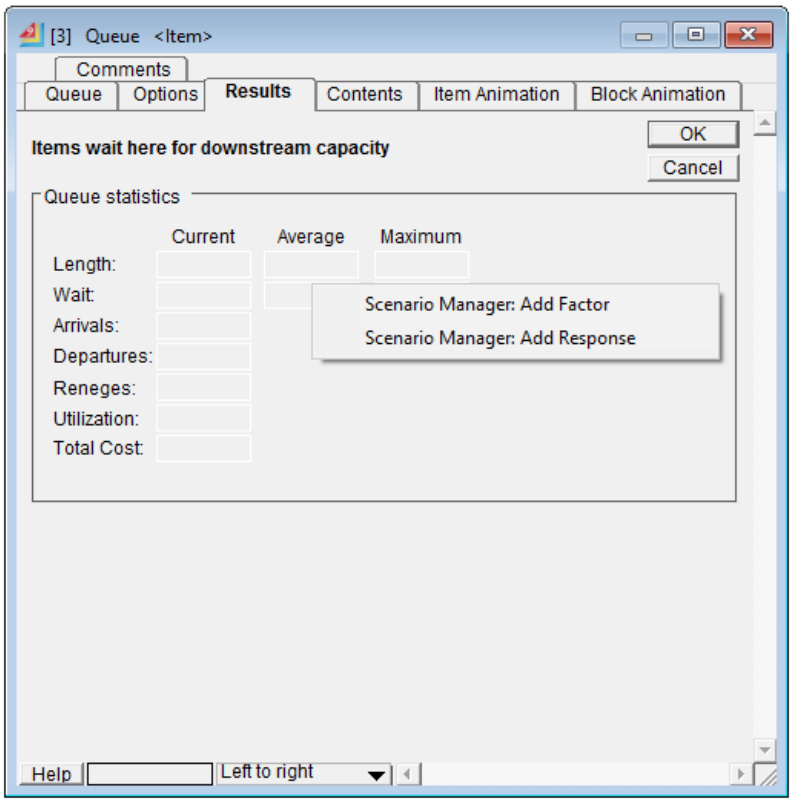

Figure 10: A Queue block window showing the option to add average Wait time as a response to the Scenario Manager. This menu is accessed by Shift+Click in the dialog box.

Clone Drop Method

1. Open the block dialog interface that contains the response needed.
2. Select the Clone arrow.
3. In a block's dialog window, drag the desired input box and drop it into the Scenario Manager's icon.
4. In the popup window that appears, select Response.
5. This value will appear in the dialog factors table in the Scenario Manager.

A step-by-step tutorial for adding dialog responses is available in the User's Guide.[4] Table 8 reviews each table input that is automatically filled. If a block is deleted but the response is still needed, the response row can simply be edited to reflect the changes in the model.

[4] *ExtendSim User Guide*. San Jose, CA (2013), Imagine That Inc., pp 639.

Table 8: Description of dialog responses table.[5]	
Table Header	**Description**
Response Name	The "Response Name" is an identifier in the Scenario results chosen by the user. It is used as a column header in the scenario table and in the report.
Block Name	The name of the block that contains the dialog variable.
Block Number	The number of the block that contains the dialog variable. The block name, label, hierarchical block name, and hierarchical block label are determined from the block number provided here. The block number is shown in the title bar of a block's dialog. In the case where the block is used inside of a hierarchical block, there are two numbers in the title bar of the block's dialog enclosed in braces, the first number is the global block number (which is used here), the second value is the block's local block number inside of the hierarchical block.
Block Label	The label of the block containing the dialog variable.
H-Block Name	The name of the hierarchical block, if the response dialog variable is in a block inside of a hierarchical block.
H-Label[num]	The local number of the block within the hierarchical block, if the response dialog variable is in a block inside of a hierarchical block.
Dialog Variable	The name of the dialog variable that is used in the scenario analysis. You can find this out by: • Using the Clone-Drop method to add the parameter to the Scenario Manager. • Performing a Shift-Right Click on the dialog item in the Scenario Manager.
Row, Column	If the parameter is in a table, enter the row and column values of that parameter in the table.
Min/Max	This value is used by JMP for scenario analysis and is only necessary if you are using JMP for analysis.
Report Set	Identifies the summary statistics that will be provided in the Scenarios table and reports. Each of the statistics will be placed in a column in the Scenarios table. The statistics that can be calculated include: Mean (M) Standard Deviation (SD) Confidence Interval (CI) - the confidence interval percentage is specified on the Scenarios Tab Variance (V) Maximum (Max) Minimum (Min) If you select "All Results" all possible statistics mentioned above are generated.
Include in Report	If checked, then the response variable will be included in reports. Sometimes it is useful to track a response variable (such as the random seed value from the Simulation Variable block), but not want to perform any statistical analysis on it.
Ignore blanks in responses (checkbox below table)	If checked, blank values for responses will not be considered in the results. For example, if there are 10 runs in a particular scenario and 2 runs have blank values for that response, only the 8 non-blank values will be used in the summary statistics calculation.

[5] Help Section of Scenario Manager Block, Imagine That, Inc.

Database Responses

Database responses can be accessed on the lower half of the Responses tab by checking the "Enter database responses" box. As with database factors, database responses are added in two ways: Manually or through the "Link" button. As the model runs, values in the source database change. At the end of each simulation run, the database variables used in the model are copied from the source to the target database variables for that scenario. This method uses a target database element, which is where response data is recorded, and the source database reference is where the data originates during the model run. Note that the target database and source database are used differently between a database factor and database response.

Manual Method

Resize the database responses table to include the number of desired database responses. Once rows are added to the table, each column's information is manually filled in. Table 9 describes each of the column's function and how the required information is inputted.

Table 9: Description of database responses table (see Figure 9).

Table Header	**Description**
Target Name	The name given to the response used as identification, typed into the cell.
Target Type	A drop down menu that allows you to select choose the target type (database, table, field, or record).
Target Database	A drop down menu that allows you to select the name of the target database. This is the database that the model results will be written to. At the end of each simulation run the database variables used in the model are copied from the source to the target database variables for that scenario.
Target Table	A drop down menu that allows you to select the name of the target table. If the target type is database, then the target table column is not used.
Target Field	A drop down menu that allows you to select the name of the target field. If the target type is database or table, then the target field column is not used.
Source Database	A drop down menu that allows you to select the database that contains the source values for the scenarios.
Source Table	A drop down menu that allows you to select the table that contains the source values for the scenarios.
Source Field	A drop down menu that allows you to select the field that contains the source values for the scenarios. If the target type is database or table, then the target field column is not used.
Source Record	A drop down menu that allows you to select the record that contains the source values for the scenarios. If the target type is database or table, then the target field column is not used.
Auto-create Target	When this is selected, the target table automatically creates a new field with the data from the source. When not selected, the target table must be already populated with the correct number of fields.
Use Factor List	Not currently used

Linking Method

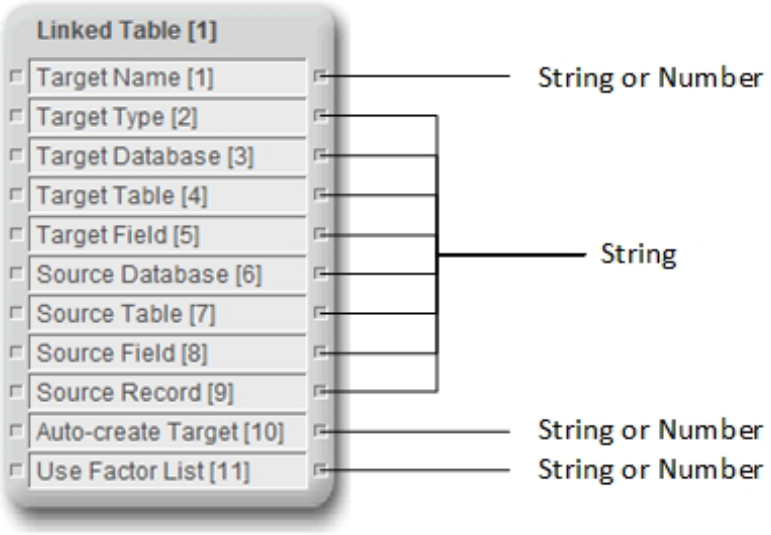

Figure 11: Field type requirements for linking a database to input database responses.

Linking the database responses table to a database table allows the user to add multiple responses simultaneously. To set up a database table to be used in the database responses table, eleven fields must be present in a table. Neither the table nor the fields have name requirements, but certain fields have field type requirements. The "Target Name," "Auto-create Target," and "Use Factor List" may be either strings or numbers, while the rest of the fields must be set as strings. For the "Auto-create Target" and "Use Factor List" columns, these records must contain any number besides zero (positive or negative) to check the box true or the number zero or any string to leave the box blank (false). All other field types must be String and must contain a valid label from the database. For instance, the second field, which corresponds to "Target Type," must contain one of the possible target types: data, table, field, or record. "Target Database," "Target Table," "Target Field," "Source Database," "Source Table," and "Source Field" must all contain the name to an actual database, table, field, and record. If a cell should be empty, then the cell may be left blank or a "-" may be used.

To delete a linked database, either resize the database responses table, or delete the table, which was linked in the database viewer.

A **database response** duplicates the source database each run and automatically names it.
A **table response** copies the source table and creates a new target for each scenario and run. The base name is chosen for each table, and each scenario and run number are included in the table name.
A **field response** copies the source field after each run and creates a new field in the target table. Each field is renamed with the scenario number and run number if the response is set to auto-create the target field automatically.

Scenarios Tab

The heart of the Scenario Manager is the Scenarios tab. Here, the modeler sets the run parameters, creates scenarios by varying factor values, runs the simulation, and monitors progress. This tab contains the values for the model factors and, after the scenarios has been run, a statistical summary of the responses. As the scenarios run, a status displays the current run count, scenario count, and the total percentage completed. The individual results for every response in every scenario run are stored in ExtendSim's graphical simulation database.

Scenarios can be created manually or by using an automatic method including a full factorial design, a JMP custom design, or Minitab optimal design. Table 10 describes the type of scenarios design options built into ExtendSim. Designs from other sources can be pasted into the Scenario table. Any number of designs can be evaluated by the Scenario Manager. The design of experiment method used is selected in the Choose DOE method drop down menu shown in Figure 13.

Table 10: Type of design options in Scenarios tab.

Type of Design	**Description**
Manually enter scenario configuration	Using the +/- data table resize button in the lower right corner of the scenarios table, add or remove rows to the scenarios table. Then the values for each input variable for each scenario can be entered. These values can be pasted in from an outside application, such as Excel.
Full factorial design	This type of analysis automatically creates a design of all possible combinations of factors.
JMP custom design	This type of analysis uses the JMP statistical analysis software. It generates a custom design with fewer experiments than the full factorial design.
Create and immediately run JMP custom design	This uses JMP to create the experiments to be run and then immediately runs all of the scenarios exporting the individual responses from each run back to JMP. If this option is selected, the Create Scenarios and Run Scenarios buttons are merged into a single button.
Minitab optimal design	Uses Minitab to add the "best" points in the candidate set sequentially. Minitab then tries to improve the initial design by adding and removing points to obtain the final design.
Create and immediately run Minitab optimal design	Uses Minitab to create the design and then immediately runs all of the scenarios exporting the individual responses from each run back to Minitab. If this option is selected, creating and running the scenarios is done in one-step so the Create Scenarios and Run Scenarios buttons are merged into a single button.

Figure 12 shows the scenario status dialog found on the Scenarios tab. In this dialog box, you set the number of runs per scenario are performed to generate the statistical information required. To determine the number of runs, statistical analysis is required. A method that can be used to identify the number of runs is covered in the "Other Considerations" portion of this document. You can also set the start time and end time for each simulation as well as the confidence interval used. This portion of the dialog box allows the user to save the model after each scenario is completed. Once the scenarios have been created, they can be run and their progress monitored in this area of the dialog box.

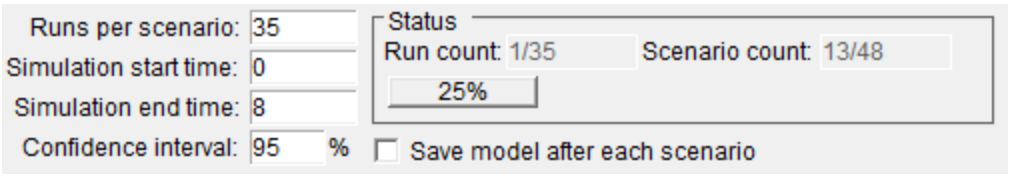

Figure 12: Scenario Manager run control and status.

The statistics displayed in the Scenarios table can be set before, after, and even during the scenarios runs. The results are displayed immediately and are updated while the model is running. These statistics may be chosen in the responses tab, or by selecting the column header of a response and choosing the desired statistics. Mean (M), standard deviation (SD), confidence interval (CI), variance (V), maximum (Max) and minimum (Min) are all available for each response based on multiple simulation runs.

Running the Model

After the scenarios are setup, the scenario run is started by pressing the Run Scenarios button in the Run Control portion of the Scenario tab in the Scenario Manager block as shown in Figure 13.

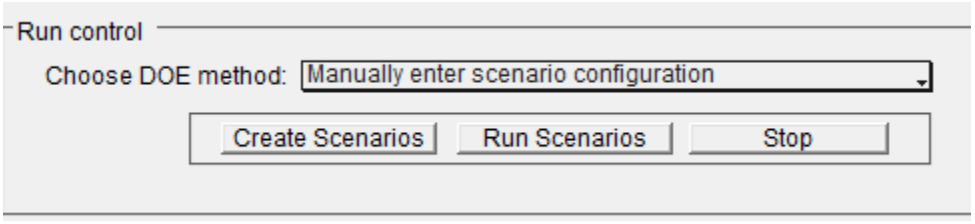

Figure 13: Run Control portion of the Scenario dialog tab.

At the start of each set of scenario runs, the Scenario Manager copies the values from the Scenarios table into the blocks and database variables from the Model inputs tab. The model is run one or more times, and the results are recorded. A summary with responses for each scenario is recorded under the Details column. There, you can see the effects of the model inputs on the model results. You can change which summary statistics are calculated in the Report set column of the Result dialog variables table.

If necessary, the scenario runs can be stopped, *the program saved*, the computer turned off, and the execution of the scenarios can be resumed later. The program is stopped by clicking the stop button in the Run Control portion of the scenario manager tab as shown in Figure 13. The Scenario Manger tracks which scenarios have been completed and will resume where the runs left off.

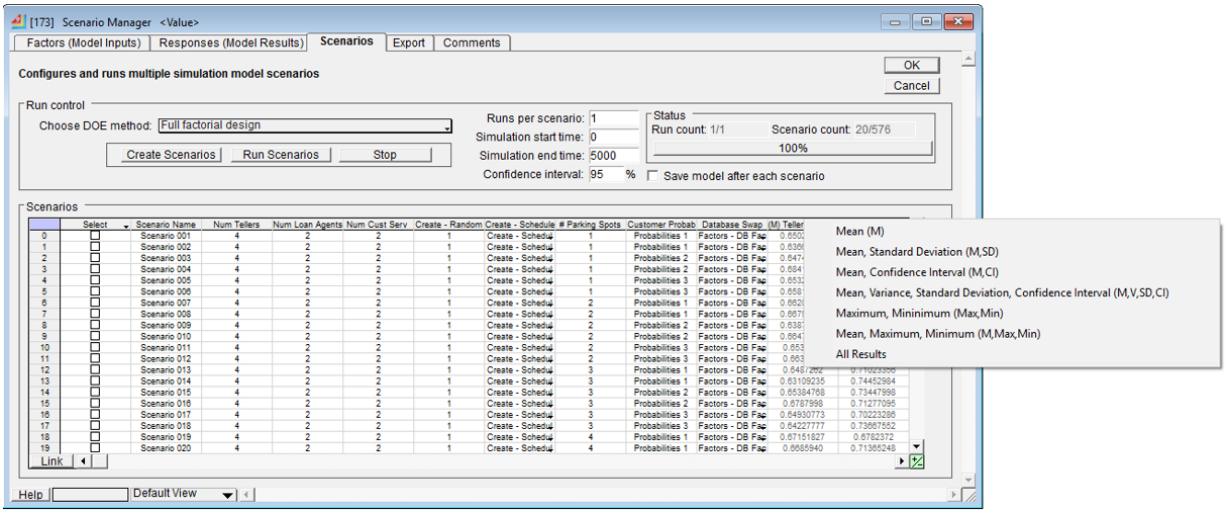

Figure 14: The Scenario Manager, Scenario tab showing scenarios population using a full factorial design. Changing response statistics can also be seen.

Export Data

The Scenario Manager is designed to work with outside applications for open-ended analysis. After all scenarios have been run, the resulting data may be exported to a database in the ExtendSim model, Excel, JMP, Minitab, or a text file under the **Export tab**. The contents of the Export Tab are shown in Figure 15.

Each scenario is exported to a separate row and the columns are labeled. Three options are available to export:

- Complete Results – Exports each scenario number, factor values and the total runs mean response values in a table. There is one row of information for each simulation run. If there are multiple runs within a scenario, there will be multiple rows of results in the report for each scenario.
- Displayed Statistics – Exports each scenario number, factor value, and mean response value as well as other statistical values currently selected in the Scenarios tab. There will be one row per scenario.
- Complete Statistics – Exports each scenario number, factor value, and all statistical results (mean, variance, standard deviation, maximum and minimum, and confidence intervals) for each scenarios response values. There will be one row per scenario.

The type of report can be chosen selecting the desired choice using the pulldown menu Report Type shown in Figure 16.

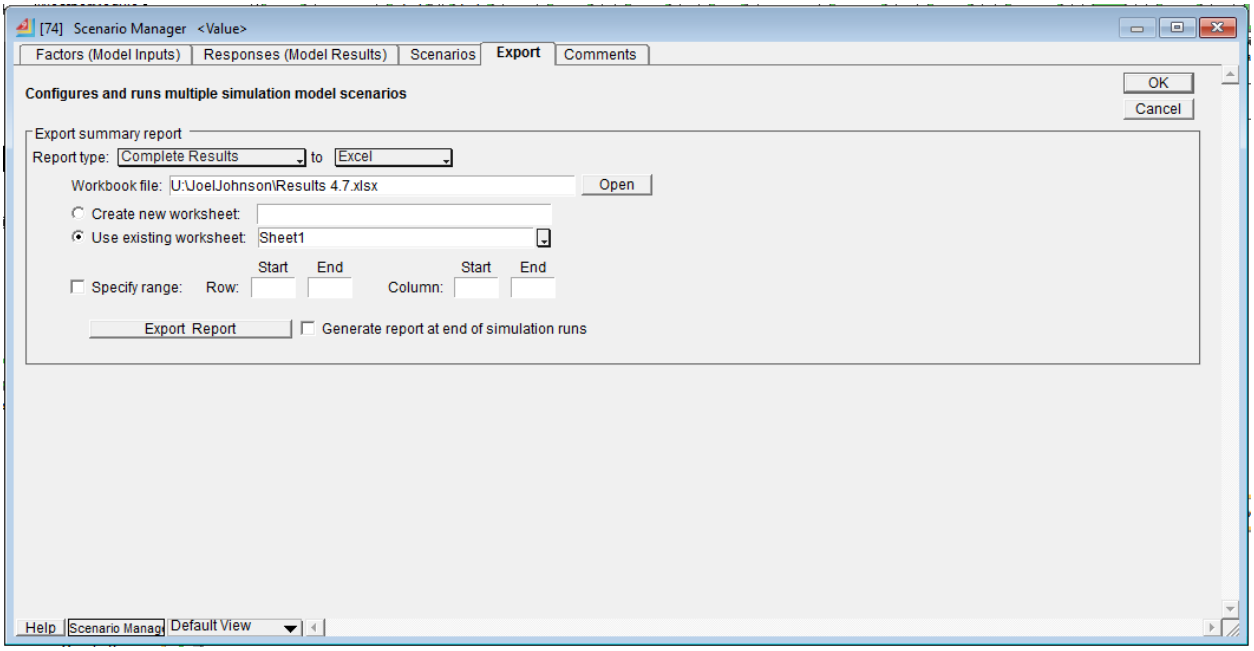

Figure 15: The export tab.

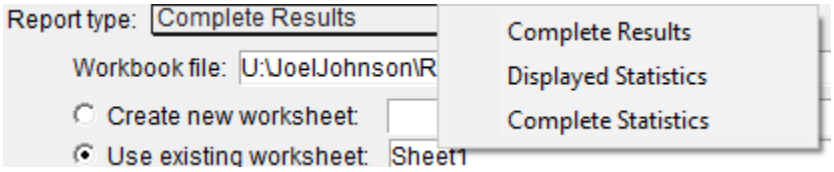

Figure 16: Options for exporting data to Excel.

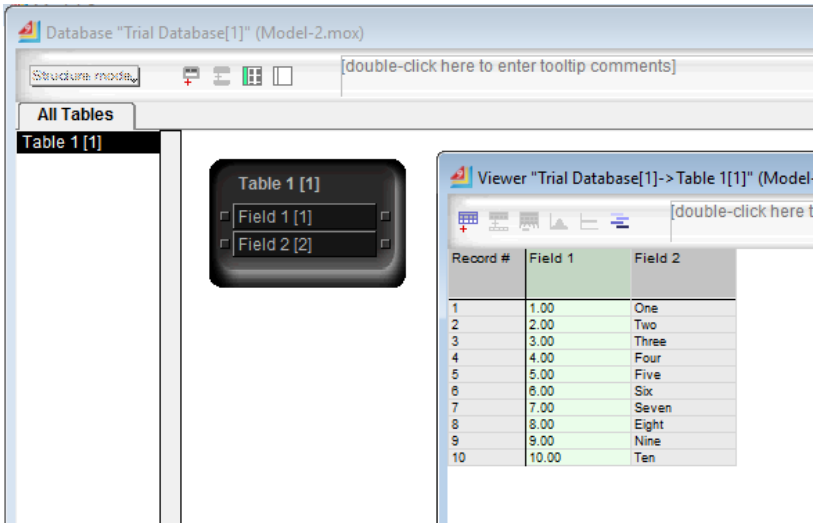

Figure 17: Example database table with 2 fields and 10 rows.

After selecting the type of report desired, the export destination must be chosen using the second drop down menu next to Report Type, also seen in Figure 15. If Database is chosen, the Export tab requests more information on the export database as shown in Figure 18. The database and table within the database can either be selected from the drop down menus or created.

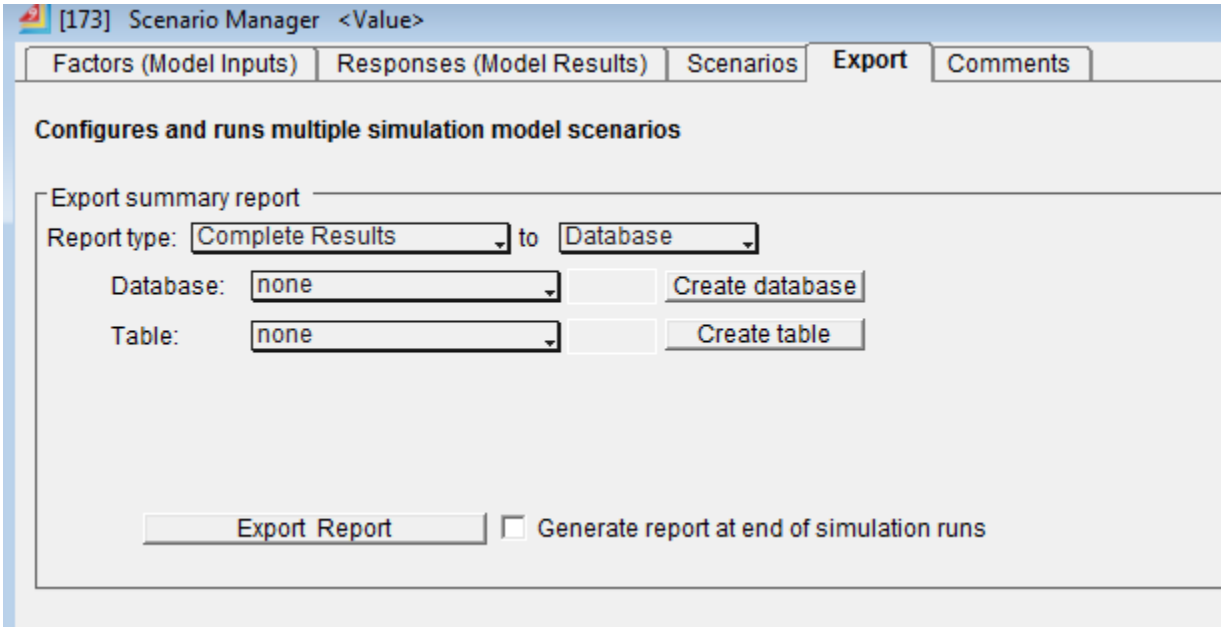

Figure 18: Export tab for export to an ExtendSim Database.

If Excel is chosen, the Export tab requests more information about the export Excel file as shown in Figure 19. The excel workbook that will contain the worksheet for export must already exist and be opened in the Export Tab. Clicking the open button allows the user to search for and select the appropriate workbook. Once the workbook is selected, the worksheet can either be selected or created. A specific range can also be selected. If no specific range is selected, the data will be exported to the worksheet starting in Column A.

Exporting to a Text file is also possible; all that is needed is a filename. Figure 20 shows the export tab for exporting to a text file. Exporting to JMP and Mintab is not discussed here since we do not often use those software packages in AET-2.

Regardless of the export destination, it is possible to export a report at the end of the simulation runs automatically by checking the "Generate report at end of simulation runs" option on the Export tab.

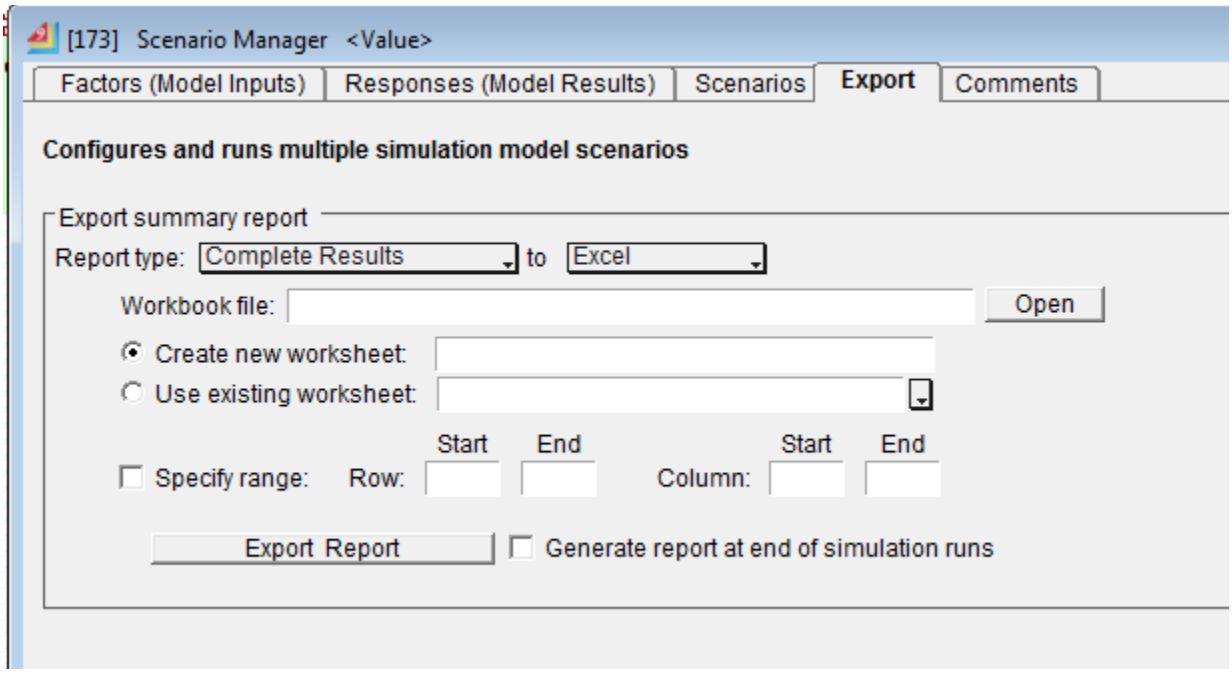

Figure 19: Export tab for export to an Excel worksheet.

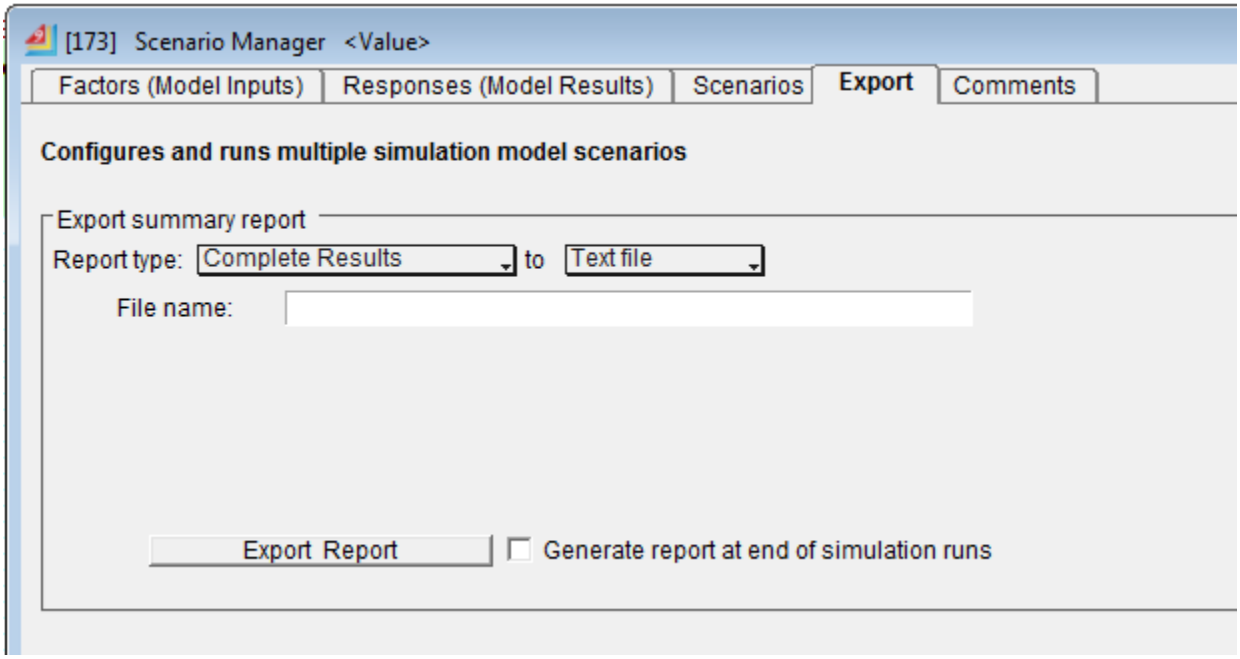

Figure 20: Export tab for exporting to a test file.

Figure 21 and Figure 22 show the result of running the model and the exported “Trial.csv” file and the format of opening it. Note that the separator between columns (currently a comma) may be changed to the user’s preference using the third section in the function `DBTableExportData` which currently contains “,”.

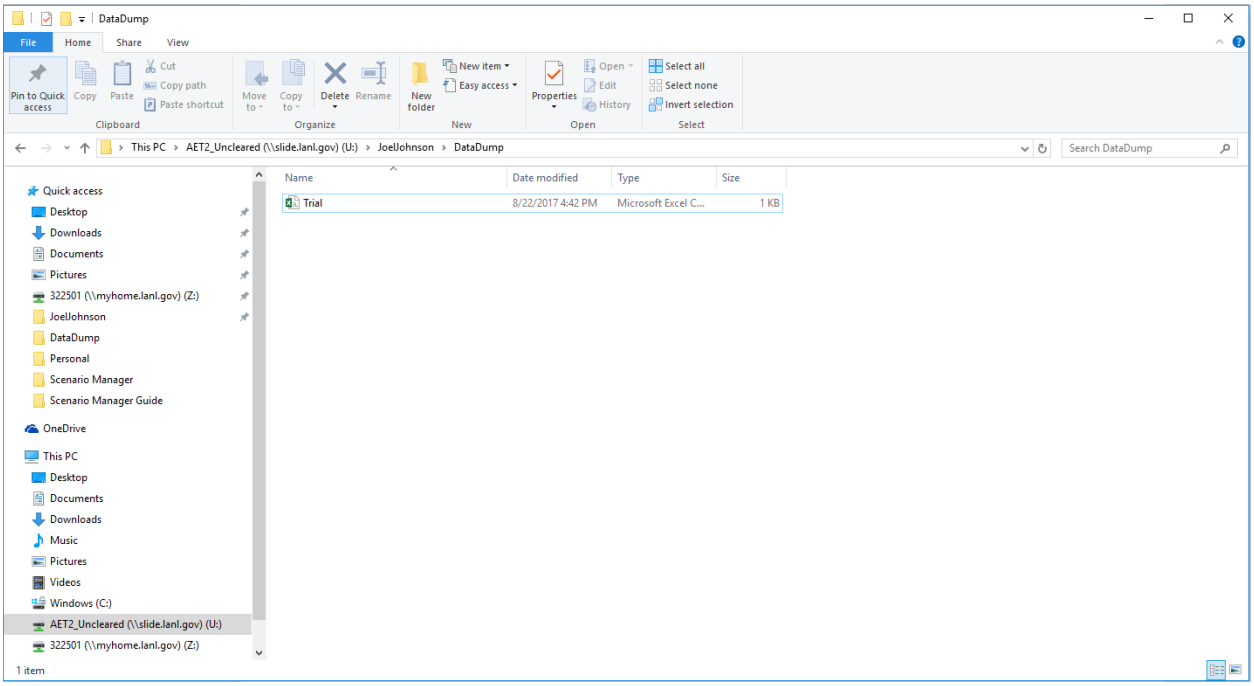

Figure 21: The folder with the resulting file.

	A	B	C
1	1	One	
2	2	Two	
3	3	Three	
4	4	Four	
5	5	Five	
6	6	Six	
7	7	Seven	
8	8	Eight	
9	9	Nine	
10	10	Ten	
11			
12			

Figure 22: Two columns and ten rows of the .csv created and opened in Excel.

Other Considerations

Determining the significant number of runs per scenario

Running a model twice will not always yield the same response values. This is due to random number generation and distributions used in a model when creating attributes, delay times, or other values.

For example, if a model uses several normal distributions to create randomness and a single scenario is run enough times with all other factors held constant, the responses themselves will form a normal distribution. As the number of runs increases per scenario, the mean converges to a point and the standard deviation of the mean (standard error) decreases. An example model was run and the mean delivery times recorded. Figure 23 shows the result of running the example model with a varying number of runs.

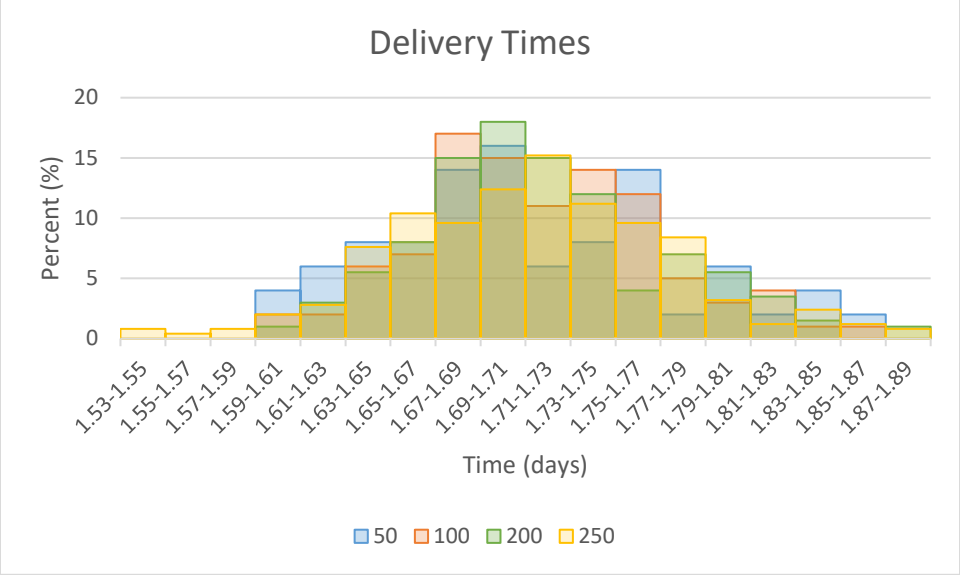

Figure 23: A response from a scenario of varying runs exhibiting a triangular distribution.

As the number of runs per scenario increases, the mean delivery time converges towards a normal distribution, and the mean converges to a value. The 95% confidence intervals are shown with error bars on each data sample. This indicates the region in which you have a 95% confidence that the mean exists. Figure 24 shows the delivery time mean for the example simulation run ten times to 250 times and displays the trend of the run mean converging to a value and the 95% confidence intervals decreasing as runs increase (as shown by the error bars).

Figure 24: Mean values as a function of number of runs per scenario. The upper and lower limits are the 95% confidence region limits (green), which in this case is 1% of the most accurate mean value (250 runs). The error bars display the 95% confidence intervals.

Figure 25 displays the standard error of the mean, as well as the 95% confidence values for each run and the upper limit for the 95% confidence levels. The trend of decreasing confidence levels with increasing runs is seen.

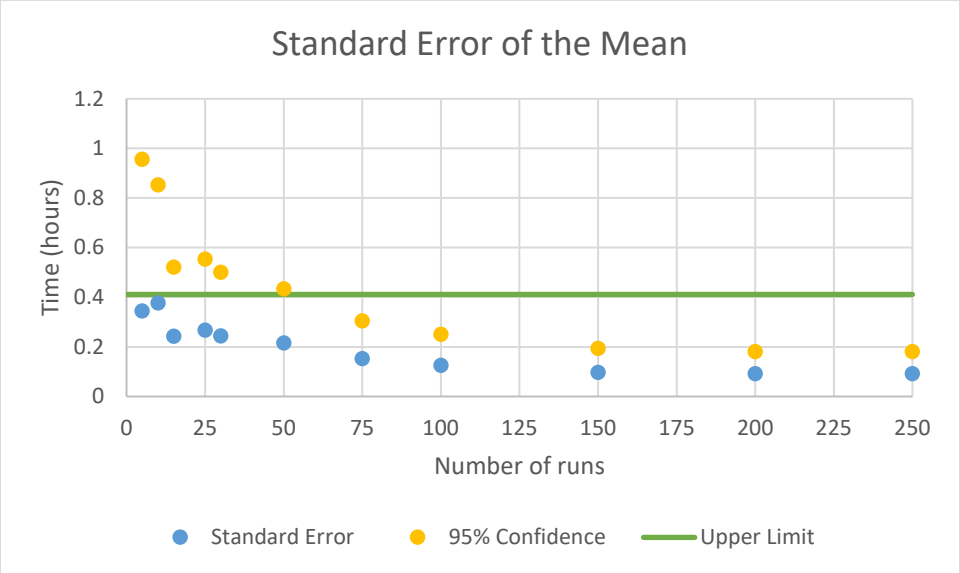

Figure 25: Standard error and 95% confidence of the delivery times as a function of number of runs per scenario.

The number of runs that should be used for any model is determined by finding the number of runs with a 95% confidence interval which is less than 1% of the mean. In this example, the mean is converging to ~41.14 hours (at 250 runs) thus the confidence upper and lower limits are 41.14 ± 0.4114. Figure 24 and

Figure 25 show that the 95% confidence is below 1% of the mean between 50 and 75 runs (approximately 55 runs).

Statistics Calculations

The mean of a set of data can be found by using Equation 1:

$$A = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad \text{Equation 1}$$

where A is the mean, N is the number of data points in the data set, and x_i is the value of each data point.

Equation 2 is used to calculate the standard deviation of the sample:

$$s = \sqrt{\frac{\sum(x - \bar{x})^2}{n - 1}} \qquad \text{Equation 2}$$

where s is the standard deviation, x represents each value in the population, $\bar{x}$ is the mean value of the sample, and n is the sample size.

With the standard deviation of the sample, the standard error of the mean may be found using Equation 3.

$$\sigma_M = \frac{s}{\sqrt{n}} \qquad \text{Equation 3}$$

where σ_M is the standard deviation of the sampling distribution of the mean, or the standard error of the mean, s is the standard deviation of the original sample, and n is the sample size.

The 95% confidence interval of the standard error can be determined using =CONFIDENCE.T(0.05,"Standard Deviation", "Sample size") function in Excel, or by using Equation 4:

$$CI = t\frac{S}{\sqrt{n}} \qquad \text{Equation 4}$$

where CI is the confidence interval, t is a critical value determined from the t_{N-1} distribution in such a way that there is area $1 - \alpha$ between t and –t, S is the standard deviation, and n is the sample size.

If data is aggregated in Excel, the functions in Table 4 may be used to calculate each statistical value.

Table 11: Excel function reference for Equation 1, Equation 2, Equation 3, and Equation 4 in order.		
Function	**Notation**	**Description**
Mean	=AVERAGE(cells)	Returns the average of its arguments
Standard Deviation of the Sample	=STDEV.S(cells)	Estimates standard deviation based on a sample
Standard Error	=(STDEV.S(cells))/(SQRT(COUNT(cells)))	No native function in Excel. Use standard deviation and count functions to implement Equation 3 into Excel which will return the standard error
Confidence Interval	=CONFIDENCE.T(1-α, s, n)	Returns the confidence interval for a population mean, using a Student's t distribution

Equation Block Export

Another method for data export from the model is by using a command line in an **Equation block**. The code command DBTableExportData(File path & name, User prompt, Format, database index, table index, rows, columns) exports a table from an internal database to a desired location and with possible file type outputs .txt, .xls, or .csv.

Example:

A model is set up with a database "Trial Database" containing "Table 1," which includes two fields, each with ten rows of data which is written during the model run. Figure 26 shows the export block setup required as a separate module in the model. A create block with an item created on a schedule at the very end of the simulation run is connected to an equation block with the code:

```
DBTableExportData("U:\JoelJohnson\DataDump\Trial.csv","","",1,1,10,2);
```

An exit block is required to complete the export module.

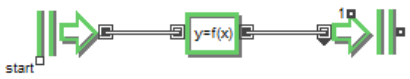

Figure 26: A simple setup example exporting data from a table in a database to a .csv.

If the run time is known the create time in the create block may reflect this. If the total run-time of the simulation is unknown, a value equation block may be added with a single out connection (outCon0) and the formula:

```
outCon0 = EndTime;
```

This assigns the delay ("D" connector) of an activity block as seen in Figure 27.

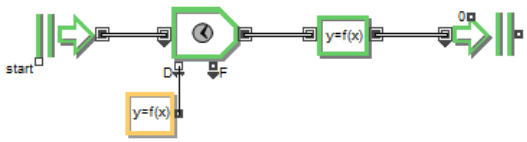

Figure 27: A modified setup example exporting data with an unknown run-time.

When run with the example database seen in Figure 17, the code is activated after an item is created and passes through the Equation block. This initiates the export command, which creates a comma-separated values file "Trial.csv" which can be opened directly in Excel.

Scenario Manager Example

Six Sigma Bank Model

The Six Sigma Bank model is a perfect simulation to illustrate how the Scenario Manager block can be used. It has multiple input parameters that can be varied to see the effect on the output parameters. The description for this model is:

> *At the Six Sigma Bank, the Six Sigma black belt would like to determine the effect that different staffing levels will have on customer service. Three different types of employees are included in this model: tellers, customer service agents, and loan agents. Customers arrive [and] must first find a parking place. Once the customer has parked their car, they will enter the bank and get in line for the type of service that they need (teller, customer service, or loan). If they want to talk to the teller, but the line is full, they…leave the bank. When the customers have completed their transaction, they will get in their car and leave. (Imagine That Inc.)*

The model has been altered slightly to include more examples of dialog and database factors/responses. Four databases have been added to the model for implementation of database factors and responses. The Create block in the original model creates item using an exponential distribution with a mean of 1 and location of 0. To illustrate table factors, the create block was changed from creating items randomly to creating items via a schedule which is read from a table in the factors database. Table 12 shows the complete list of inputs and outputs which are used as examples in the scenario manager.

Table 12: Six Sigma Bank Model input parameters and outputs.

Type	Block type	Description	Implementation in Scenario Manager
Input	Constant	Number of tellers	Dialogue factor
Input	Constant	Number of loan agents	Dialogue factor
Input	Constant	Number of customer service agents	Dialogue factor
Input	Create	Scheduled item generation	Database factor – Table
Input	Resource Pool	Number of parking spaces	Database factor – Record
Input	Random Number	Customer creation probabilities	Database factor – Field
Input	-	Swap the entire Factor database with other databases	Database factor – Database
Output	Queue	Teller utilization	Dialog response
Output	Queue	Loan agent utilization	Dialog response

Output	Queue	Customer service agent utilization	Dialog response
Output	Queue	Teller agent queue average wait	Dialog response
Output	Queue	Loan agent queue average wait	Dialog response
Output	Queue	Customer service agent queue average wait	Dialog response
Output	Statistics	All model data exported to a table	Database response – Table
Output	Exit	Total customers exited	Database response – Field

A blank model for practice may be accessed in the ExtendSim examples folder (Examples\Discrete Event\Bank.mox). A blank model with databases implemented can be found in the same folder as this guide is found (or by contacting the author). With the factors and responses found in Table 12 in the bank model, a full analysis may be completed with each scenario created from the factors. If a full factorial design is performed, 576 scenarios will be created.

For additional information, please contact
Joel Johnson
Process Modeling and Analysis Group (AET-2)
P. O. Box 1663, MS E548
Los Alamos National Laboratory
Los Alamos, NM 87545
505-667-7097
joeljohnson@lanl.gov